Poster: Design of Survivable SDN/NFV-enabled Networks with Bandwidth-optimal Failure Recovery

Andrea Tomassilli, Giuseppe Di Lena, Frédéric Giroire, Issam Tahiri, Stéphane Pérennes, Damien Saucez, Thierry Turletti, Ruslan Sadykov, François Vanderbeck, Chidung Lac

Poster: Design of Survivable SDN/NFV-enabled Networks with Bandwidth-optimal Failure Recovery

A. Tomassilli*, G. Di Lena§†, F. Giroire*, I. Tahiri‡, D. Saucez§,
S. Perennes*, T. Turletti§, R. Sadykov‡, F. Vanderbeck‡ and C. Lac†
*Université Côte d'Azur, CNRS, Inria Sophia Antipolis, France
‡Université de Bordeaux, Institut de Mathématiques, France
§Inria, Sophia Antipolis, France
†Orange Labs, France

***Abstract*—ISP networks are taking a leap forward thanks to emerging technologies such as Software Defined Networking (SDN) and Network Function Virtualization (NFV). Efficient algorithms considered too hard to be put in practice on legacy networks now have a second chance to be considered again. In this context, we rethink the ISP network dimensioning problem with protection against Shared Risk Link Group (SLRG) failures. We consider a path-based protection scheme with a global rerouting strategy in which, for each failure situation, we may have a new routing of all the demands. Our optimization task is to minimize the needed amount of bandwidth. We develop a scalable mathematical model that we handle using the Column Generation technique. We show the effectiveness of our methods and demonstrate the feasibility of our approach using Mininet.**

I. Introduction

The design of survivable networks is one of the fundamental problems in networking [1]. Its goal is to design the cheapest network in terms of resources that can satisfy the set of estimated traffic demands, while allowing to recover from failures. The emergence of new network paradigms such as Network Function Virtualization (NFV) and Software Defined Networking (SDN), as well as the development of new networks, e.g., for 5G and IoT, lead to the reconsideration of traditional network design. Faults in the IP and optical layers tend to be correlated [2]. To model this correlation, *Shared Risk Link Groups* (SRLGs) have been introduced [3]. SRLGs allow to easily express a risk relationship. We consider a protection technique called *unrestricted flow reconfiguration*, also known as *global rerouting* [1]. In each of the possible failure situations, a new set of backup paths are defined, one for each demand. This makes this protection method *bandwidth-optimal*. Our goal is to compute for each demand *a primary and a backup path for each SRLG failure scenario*, while ensuring that the required network functions will be performed on the packets in the order specified by its Service Function Chain (SFC).

II. Problem Statement and Notations

We model the network as an undirected graph $G = (V, E)$, where V represents the set of nodes and E the set of links. We are given a set of SRLG events $\mathcal{R}$ that can incur link failures. Each $r \in \mathcal{R}$ consists of a set of links that share a common physical resource. We denote by $\mathcal{D}$ the set of demands. A demand $d \in \mathcal{D}$ is modeled by a quadruple (s_d, t_d, bw_d, C_d) with s_d the source, t_d the destination, and C_d the ordered sequence of network functions that need to be performed. Given the network topology and the traffic rate of the demands to be supported, the purpose of the design problem is to precompute a set of paths to guarantee the recovery of all the demands in the event of an SRLG failure, while satisfying their SFC requirements. The considered optimization task is to *minimize the required bandwidth in the network*.

III. Optimization Approaches

We start the section by proving hardness and inapproximability results for the Global Rerouting problem. Then, we propose a scalable decomposition model which relies on the Column Generation technique.

Proposition. 1. *The* Global Rerouting *problem is NP-hard even for a single demand, and cannot be approximated within* $(1-\epsilon)\ln(|R|)$ *for any* $\epsilon > 0$ *unless P=NP, where* $|R|$ *denotes the number of failing scenarios.*

We denote by Π_d^r, the set of service paths for a demand d in the SRLG failure situation r. Each service path π is associated with an integer value $a_{uv}^{\pi} \geq 0$ telling the number of times the link (u, v) is used in the service path π. Variables are as follows.

- $y_{\pi}^{d,r} \geq 0$, where $y_{\pi}^{d,r} = 1$ if demand d uses path π as a service path in the SRLG failure event $r \in \mathcal{R}$.
- $x_{uv} \geq 0$, is the bandwidth allocated on link $(u, v) \in E$.

Objective: minimization of the required bandwidth

$$\min \sum_{(u,v) \in E} x_{uv}. \tag{1}$$

One service path for each demand and SRLG failure event: for all $d \in \mathcal{D}$, $r \in \mathcal{R}$

$$\sum_{\pi \in \Pi_d^r} y_{\pi}^{d,r} \geq 1. \tag{2}$$

Bandwidth utilization: for all $(u, v) \in E$, $r \in \mathcal{R}$

$$x_{uv} \geq \sum_{d \in \mathcal{D}} \sum_{\pi \in \Pi_d^r} bw_d \cdot a_{uv}^{\pi} \cdot y_{\pi}^{d,r}. \tag{3}$$

At each iteration, the restricted master program (RMP) is solved. The dual values associated to the constraints are used to define new paths with negative reduced cost to be added to the RMP that may enable. This process is repeated until no

Network	z^*_{LP}		MasterILP		IterILP		IterRR	
	ColGen	Benders	time	ϵ	time	ϵ	time	ϵ
pdh	22s	32s	11mn	4%	1mn	4.82%	40s	12.7%
polska	15s	18s	40s	0.22%	1mn	0.1%	20s	1.4%
nb-germany	35s	1mn	40s	0.17%	4mn	0.06%	30s	3.2%

TABLE I: Numerical results for the optimization models. The first column refers to the time needed to find the optimal fractional solution with the Column Generation and the Benders Decomposition approaches.

more columns with negative reduced cost exist. Associated to the optimal solution of the linear relaxation of the RMP, for each demand d and SRLG failure situation r, there is a set of service paths that guarantee the minimum cost in terms of required bandwidth to deploy in order to guarantee the recovery in the splittable flow case. However, if we restrict our attention to the unsplittable flow case, we have to select only one service path for each demand and SRLG failure situation. The problem now consists in making this choice by reducing the *overflow* introduced in the network.

One possible way consists in changing the domain of the variables in the last RMP from continuous to integer and use an integer linear program (ILP) solver. We refer to this strategy as MasterILP. Another approach may consist in efficiently computing a fractional solution to the linear relaxation of the problem (i.e., when flows are splittable) and trying to obtain a *good* integer solution to the problem (i.e., when flows are unsplittable) by minimizing the cost to pay in terms of additional capacity (i.e., the *overflow*) over all the scenarios. We define overflow as the total amount of additional bandwidth to be allocated in the network in order to satisfy all the demands. One possible strategy to do that may consist in considering each scenario one at a time, and formulating a multicommodity flow problem as an ILP. The objective function consists in minimizing the overflow to be allocated in the network. We refer to this strategy as IterILP.

Another strategy consists in using an algorithm to route the demands while minimizing the overflow.

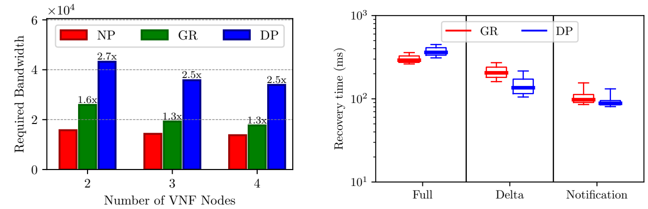
Proposition. 2. *The* MIN OVERFLOW PROBLEM *is APX-hard and cannot be approximated within a factor of* $1 + \frac{3}{320}$, *unless P=NP.*

Proposition. 3. *The* MIN OVERFLOW PROBLEM *can be approximated with high probability within a factor of* $(1 + \frac{1}{e}) + \epsilon$, *for any* $\epsilon > 0$.

IV. NUMERICAL RESULTS

In this section, we evaluate the performance of our proposed algorithms. We conduct experiments on three real-world topologies from SNDlib [4]: `polska`, (12 nodes, 18 links, and 66 demands), `pdh` (11 nodes, 34 links, and 24 demands) and `nobel-germany` (17 nodes, 26 links, and 121 demands). In Table I, we show the performances of our proposed algorithms on the selected networks.

Now, we discuss how to implement our global rerouting proposition with OpenFlow and we evaluate it with the Mininet emulator [5] and OpenDayLight controller. Our evaluation in realistic conditions shows that implementation choices have a significant impact on the recovery time of protection mechanisms. A first option to implement the protection scheme in OpenFlow is to let the SDN controller fully update the flow tables on the switches upon failure. When the controller detects a failure, it sends the new flow tables to the impacted switches; we refer to this option as *full*. A variation of this option is to only send the changes to be performed on the flow tables to the switches to reduce the signaling load and the number of flow table updates on the switches. We name this option *delta*. Another option is to leverage the Multiple Flow Tables capability introduced in OpenFlow 1.3 to pre-install the flow tables for each SRLG failure scenario in the switches. When the controller sends a failure notification to a switch, the switch activates the appropriate flow table in only one operation (using goto), this option is called *notification*. In Fig. 1, we show both the bandwidth overhead for different protection schemas (Fig. 1.a) and the *recovery time* (Fig. 1.b) defined as the span of time between a failure event and the moment in which all switches are updated to be in a state that circumvents the failure.



(a) Bandwidth Overhead (b) Recovery Time

Fig. 1: Left: Bandwidth overhead comparison of the global rerouting (GR) and Dedicated Path Protection (DP) schemas with respect to the no-protection scenario (NP). Right: Recovery time for the different protection schemas for the polska network.

V. CONCLUSION

We studied the ISP network dimensioning problem with protection against a Shared Risk Link Group failure. We considered a path-protection method based on a global rerouting strategy, which makes the protection method optimal in terms of bandwidth. We proposed algorithms to compute the backup paths for the demands which rely on the Column Generation technique. We validated them and we showed the applicability of the global rerouting protection method thanks to SDN with a real implementation using the OpenDaylight controller.

REFERENCES

[1] M. Pióro and D. Medhi, *Routing, flow, and capacity design in communication and computer networks*. Elsevier, 2004.

[2] S. Kandula, D. Katabi, and J.-P. Vasseur, "Shrink: A tool for failure diagnosis in ip networks," in *Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data*. ACM, 2005, pp. 173–178.

[3] D. Papadimitriou, "Inference of shared risk link groups," *Internet-draft: draft-many-inference-srlg-02. txt*, 2001.

[4] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski, "Sndlib 1.0—survivable network design library," *Networks*, vol. 55, no. 3, 2010.

[5] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: Rapid prototyping for software-defined networks," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*. ACM, 2010.